

### Abstract of the Disclosure

An improved Flash memory device with a distributed erase block management (EBM) scheme is detailed that enhances operation and helps minimize write fatigue of the floating gate memory cells of the Flash memory device. In the prior art, erase block management of a Flash memory device, which provides logical sector to physical sector mapping and provides a virtual rewriteable interface for the host, requires that erase block management data be kept in specialized EBM data tables to keep the state of the Flash memory device in case of loss of power. This placement of EBM data in a separate erase block location from the user data slows the Flash memory operation by requiring up to two writes and/or block erasures for every update of the user data. Additionally, one of the goals of the EBM control is to minimize write fatigue of the non-volatile floating gate memory cells of the Flash memory device erase blocks by re-mapping and distributing heavily rewritten user data sectors in a process called load leveling so that no one erase block gets overused too quickly and reduce the expected lifespan of the Flash memory device. The EBM data structures, however, are some of the most heavily rewritten non-volatile floating gate memory cells in the device and thus, while helping to reduce write fatigue in the Flash memory device, are some of the data structures most susceptible to the process of fatigue. The Flash memory device of the invention combines the EBM data in a user data erase block by placing it in an EBM data field of the control data section of the erase block sectors. Therefore distributing the EBM data within the Flash memory erase block structure. This allows the Flash memory to update and/or erase the user data and the EBM data in a single operation, to reduce overhead and speed operation. The Flash memory also reduces the process of EBM data structure write fatigue by allowing the EBM data fields to be load leveled by rotating them with the erase blocks they describe.